

**System and Method for Dynamic Price Determination
in Differentiated Services Computer Networks**

RELATED APPLICATIONS

This application is related to the following co-
5 pending U.S. Patent Application each filed on March 15,
2001 and each having the same inventors and assignee:
"System and Method for On-Demand Pricing for Differentiated
Services Computer Networks" (Serial No. 09/810,063); and
10 "System and Method for Pricing Agent for Differentiated
Services Computer Networks" (Serial No. 09/810,062), each
by Grande and Kouadio, and each assigned to International
Business Machines Corporation.

BACKGROUND OF THE INVENTION

1. Technical Field

15 The present invention relates in general to a system
and method for providing network usage pricing based on a
user's service demands. More particularly, the present
invention relates to a system and method for providing
server based technology to provide clients with pricing
20 information and maintain pricing information for billing
purposes.

2. Description of the Related Art

Computer systems in general and International Business
Machines (IBM) compatible personal computer systems in
25 particular have attained widespread use for providing
computer power to many segments of today's modern society.
Systems with microprocessors are finding themselves in an

0989437-062601
TOP SECRET

array of smaller and more specialized objects that previously were largely untouched by computer technology. Computer systems typically include a system processor and associated volatile and non-volatile memory, a display area, input means, and often interfaces, such as a network interface or modem, to other computing devices.

One of the distinguishing characteristics of these systems is the use of a system board to electrically connect these components together. These computing devices are information handling systems which are designed primarily to give independent computing power to a single user, or a group of users in the case of networked computing devices. Personal computing devices are often inexpensively priced for purchase by individuals or businesses. Nonvolatile storage devices such as hard disks, CD-ROM drives and magneto-optical drives are considered to be peripheral devices. Computing devices are often linked to one another using a network, such as a local area network (LAN), wide area network (WAN), or other type of network, such as the Internet. By linking to other computer systems, a computing device can use resources owned by another computing device. These resources can include files stored on nonvolatile storage devices and resources such as printers.

Servers perform different services for client computer systems. Web servers often provide content, or information, to client computers. Content provided by servers varies widely and includes such things as simple textual information (such as news articles and stock quotes), multimedia information (such as sound and

graphics), software that is either downloaded to the client computer or executed to some extent on the server computer, and even real-time audio and video (such as that used in video conferencing). In order to run efficiently, some content needs to be provided to the user in a timely basis (such as a video conferencing application), while other content, such as text articles, is not compromised if the information is not provided on an expedited basis.

In order to accommodate users' needs, a differentiated services Internet is being developed to prioritize packets of information that need to be delivered to the user on a faster basis. Packets include a header portion that determines, among other things, the destination of the packet (i.e., a server or client computer address). In a differentiated services Internet, the packets further include prioritization information detailing whether the packet is a high or low priority packet. High priority packets, such as those containing real-time teleconferencing information, are handled differently than lower priority packets. As the packets travel through the Internet from one computer to another, they pass through other computers including specialized devices called routers. Routers and other points along the Internet can be programmed to treat high priority packets differently in such a way that those packets travel through the Internet faster than their lower priority counterparts.

Users, often referred to as clients, generally connect to the Internet using an Internet Service Provider (ISP). While some plans bill the user a connection fee for the amount of time the user is connected to the Internet, most

plans generally provide for flat based pricing whereby the user has either unlimited connection or a given amount of connect time for a flat periodic (i.e., monthly) charge. In addition, more users are using broadband services, such as cable modem connections, digital subscriber line (DSL) connections, and integrated services digital network (ISDN) connections. Besides offering faster transmission speeds than found with traditional modems, these connections often provide a constant connection between the user's computer and the Internet whenever the user's computer is turned on. In addition, while broadband connections provide for faster transmission speeds between the user's ISP and the user's computer, they do not effect the prioritization or treatment of the user's packets once they have been transferred from the ISP to various routers on their way to the server.

A challenge, therefore, exists with users use of differentiated services on the Internet. Users with high priority requirements, such as video conferencing, need a way to switch between high and normal priorities. A further challenge is charging high priority users a premium for their increased and preferential usage of Internet resources.

In addition, pricing for differentiated services may change. Users may be reluctant to purchase priority service without knowing beforehand what they will be charged for using the service. Another challenge, therefore, exists in informing users of the current priority pricing before the user establishes a priority service connection.

[illegible]

SUMMARY

It has been discovered that a server can determine pricing for clients' high priority usage of a network by evaluating the network load or using other factors such as the time of day or day of the week. Clients (i.e., users) request pricing information from a server, such as a Internet Service Provider (ISP). The server has access to pricing information that is retrieved and returned to the clients.

Users can request high priority treatment which causes packets sent to and from the user to be marked, or flagged, as high priority packets. The amount of time the user is operating as a high priority user is recorded so that the user can be billed accordingly. When the user requests high priority treatment, the server (ISP) marks the time that the user began using the high priority service and the price that was either quoted by the server or that was current at the time the request was made. When the user no longer wishes to operate in a high priority mode, he can turn off the service upgrade request at which time the user's packets are marked as normal priority packets to which flat rate or normal billing rates apply. The server intercepts the client's request to return to normal priority service and marks the billing file accordingly. At the end of a billing cycle, the user's Internet Service Provider calculates the amount of time the user spent as a high priority user and determines a price by combining the normal (or flat) rate changes with the high priority charges incurred by the user.

When computer systems or routers receive high priority packets destined to or from the user, the packets are handled in a manner to allow the packets to pass more quickly through the Internet on their way to or from the user. For example, devices such as routers include queues containing the packets currently being transmitted through the router. During busy traffic periods, the router may become full necessitating the router to drop certain packets. Packets marked as high priority packets, however, are not dropped thus increasing high priority packet throughput with respect to normal priority packets. In addition, routers can be programmed to process high priority packets before normal priority packets, even though the high priority packet may arrive at the router after the normal priority packet.

Internet devices, such as routers, are polled by computing devices, such as ISPs or servers, to determine the amount of traffic on the computer network. Higher service upgrade prices can therefore be applied when it is determined that the network has a high amount of traffic and lower service upgrade prices can be applied when traffic on the network is lower. The determined price can either be applied to all the time the client spends using an upgrade session or the price may be changed during the session as traffic demands on the network increase or decrease. In addition, if a variable pricing scheme is used the client can request to be informed of price changes and may additionally set a threshold to be informed only when the change reaches a certain percentage level.

The foregoing is a summary and thus contains, by necessity, simplifications, generalizations, and omissions of detail; consequently, those skilled in the art will appreciate that the summary is illustrative only and is not intended to be in any way limiting. Other aspects, inventive features, and advantages of the present invention, as defined solely by the claims, will become apparent in the non-limiting detailed description set forth below.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention may be better understood, and its numerous objects, features, and advantages made apparent to those skilled in the art by referencing the accompanying drawings. The use of the same reference symbols in different drawings indicates similar or identical items.

Figure 1 is a system diagram showing a computer system using high and normal priorities;

Figure 2 is a block diagram of a packet traveling between a client computer system and a server computer system through various routers;

Figure 3 is a flowchart of service upgrade and billing processes for handling high priority packets;

Figure 4 is a flowchart for router processing;

Figure 5 is a block diagram of a client computer system using a pricing daemon to notify the user of the current service upgrade price;

Figure 6a is a window interface for a user accepting or rejecting a service upgrade;

Figure 6b is a flowchart for daemon processing for accepting or rejecting service upgrade pricing;

Figure 7 is a block diagram of a server computer system using processes to determine pricing levels, respond to client inquiries, and track client service level changes;

5 **Figure 10** is a block diagram of an information handling system capable of implementing the present invention.

Station	Time	Lat	Long	Alt	Temp	Hum	Wind	Dir	Speed	Clouds	Vis	Remarks
1	0000	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
2	0100	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
3	0200	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
4	0300	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
5	0400	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
6	0500	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
7	0600	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
8	0700	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
9	0800	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
10	0900	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
11	1000	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
12	1100	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
13	1200	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
14	1300	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
15	1400	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
16	1500	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
17	1600	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
18	1700	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
19	1800	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
20	1900	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
21	2000	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
22	2100	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
23	2200	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear
24	2300	34 15 N	121 05 E	10	10.0	85	10	000	000	000	10	Clear

DETAILED DESCRIPTION

The following is intended to provide a detailed description of an example of the invention and should not be taken to be limiting of the invention itself. Rather, any number of variations may fall within the scope of the invention which is defined in the claims following the description.

Figure 1 shows a system diagram of a computer system using high and normal priorities. Client computer system **100** is connected to computer network **110**, such as the Internet. Client computer system **100** can be connected to computer network **110** in a variety of ways. For example, client computer system **100** can use a modem to dial into an Internet Service Provider (ISP) that allows client computer system **100** access to computer network **110**. Client computer system **100** could also use a cable modem connection, a DSL connection, a satellite connection, or an ISDN connection to access computer network **110**. The user of client computer system **100** can choose to use higher speed service or lower speed service. When the user chooses to use higher speed service, then higher speed packets **120** are sent to and from client computer system **100**. During normal operations, when the user does not need higher speed service, lower speed packets **130** are sent to and from client computer system **100**. The priority of the packets is determined by analyzing the header area included with the packets. Among other things, the header area determines the destination of the packet and the priority assigned to the packet. Both higher speed packets **120** and lower speed packets **130** travel across computer network **110** to a

destination computer system, such as server computer system
150. Server computer system 150 receives request packets
140 from various client computer systems, including client
computer system 100, processes the requests, and returns
5 response packets 160 to the client computer systems that
made the corresponding requests. When a request packet
includes a higher priority header, the server includes the
higher priority header information in the corresponding
response packet. In this manner, the higher priority
10 packets both arrive at the destination computers in a
prioritized fashion and are returned to the requesting
client computer in a prioritized fashion. The return trip
back to the client computer is often more important for
prioritized processing because the client computer is often
15 requesting large data files or data streams, such as those
including multimedia content. These larger files benefit
from prioritized processing in that larger files generally
take longer to transmit over the Internet, especially
during peak usage periods. However, with prioritized
20 processing, the user is able to receive the content within
a needed timeframe. To keep track of the user's
prioritized processing, billing computer system 170
monitors the traffic to and from client computer system
100. In order to monitor the traffic, billing computer
25 system 170 may be incorporated within the client's ISP.
Packets traveling to and from the client computer system
pass through the ISP on their way to or from the Internet.
Billing computer system 170 analyzes the packet headers to
determine which client computer is sending or receiving the
30 packets. When prioritized header information is included
in the packets, billing computer system 170 records the
client's use of prioritized service in billing records data

store 180. Various billing plans can be established for use of prioritized service. For example, users could purchase a certain amount of prioritized service per month as part of their regular Internet service bill, with additional prioritized service being charged on a per-minute basis. At the end of a billing cycle, billing computer 170 reads the recorded billing records 180 and prepares a network usage bill 190 for each of the ISP's customers.

Figure 2 shows a block diagram of a packet traveling between a client computer system and a server computer system through various routers. In the example shown, three routers (router 220, router 240, and router 260) are used to route packet 210 between client computer system 200 and server computer system 280. Packet 210 includes a header area and a data area. The header area indicates the source and destination addresses for the packet and any prioritization information. The data area of packet 210 includes the content being transmitted between the client and the server. The content may be a request (i.e., from the client to the server) or a response to a request (i.e., from the server to the client). Each router includes a queue identifying packets that have been received and that are waiting to be forwarded in order to reach their ultimate destination. Router 220 includes queue 230, router 240 includes queue 250, and router 260 includes queue 270. At each of the routers, packet 210 is added to the corresponding router queue. Each router includes a queue handling routine to process items in its queue. To provide prioritized service, the queue handling routine in each router reads the header portion of each packet to

identify high priority packets. If a high priority packet is identified, logic within the queue handling routine is invoked to provide priority service to the identified packet. One priority service that the queue handling

5 routine can provide regards packet dropping. When a queue becomes too busy and the memory allocated for the router's queue becomes full, some packets need to be dropped from the queue. When a packet is dropped, either client **200** or server **280** determines that a packet is missing and requests

10 the other device to retransmit the missing packet. Consequently, retransmission of packets causes an increased in the overall transmission time needed to transmit the packet. The queue manager routine can avoid dropping high priority packets and, instead, drop lower priority packets

15 waiting on the queue. Another way the queue manager routine can increase the throughput of high priority packets is by reordering the queue to put high priority packets toward the top of the queue. One type of reordering routine uses the amount of time a packet has

20 waited on the queue and the packet's priority to determine a queue order. Those packets who have waited longer and those packets with higher priorities are thus moved toward the top of the queue, while packets that have newly entered the queue and packets with lower priorities are moved

25 towards the end of the queue.

Figure 3 shows a flowchart for service upgrading and billing processes for handling high priority packets. The processing could be part of an ISP's handling of client requests. In this way, the user would send a message to

30 the ISP requesting a service upgrade in order to improve throughput to and from the client's computer system.

Processing commences at **300** whereupon a determination is made as to whether the user has requested a service upgrade (decision **310**). If the user has not requested a service upgrade, "no" branch **315** is taken. During normal processing, packets are marked with a default priority setting (step **320**) and the user is billed at the normal or flat rate (step **330**) for handling packets. If the normal billing involves an hourly or per-minute rate, the customer's billing records are updated to track the amount of time that the customer spent using the network.

On the other hand, if the customer requested a service upgrade, decision **310** branches to "yes" branch **335**. During a service upgrade, the header area of the customer's packets are marked to reflect the higher priority setting (step **340**). The customer's use of the service upgrade is recorded in the customer's billing records (step **350**). The higher priority packet marking and service upgrade charges continue until the customer requests that the service upgrade period be stopped or when the user signs off.

20 In either case, after the packets are marked and the appropriate information is captured in the billing records, the customer sends request packets and receives responsive packets from servers or other content providers (step 360). A determination is made as to whether the user has signed off (decision 370). If the user has not signed off, "no" branch 375 is taken looping back to decision 310 to check whether the user is requesting a service upgrade. This looping continues until the user requests to sign off, at which time "yes" branch 385 is taken and processing ends at 390.

Figure 4 shows a flowchart for router processing of high priority packets across the Internet. The router processing shown could be incorporated in the router's queue manager routine (discussed in **Figure 2**) for handling packets queued at the router.

Processing commences at **400** whereupon a determination is made as to whether the memory allocated for the router's queue is full (decision **410**). If the router's queue is full, decision **410** branches to "yes" branch **415** wherein the packets at the end of the queue are read (input **420**). A determination is made as to whether each packet is a high priority packet (decision **430**). If the packet is not a high priority packet, then "no" branch **435** is taken whereupon the low priority packets are dropped from the queue (step **440**) and the memory associated with the dropped packets is freed for more queue entries. On the other hand, if packets read from the queue are high priority packets, then "yes" branch is taken preventing the high priority packets from being dropped from the queue. Processing loops back to the beginning (loop **450**) to continue managing the router's queue.

If the memory allocated for the router's queue is not full, decision **410** branches to "no" branch **455** whereupon a determination is made as to the amount of congestion, or traffic, within the router (decision **460**). If traffic is high, it may make sense to organize the queue in such a fashion to provide greater throughput for high priority packets. In such a case, "yes" branch **465** is taken and the queue is reorganized in order to move high priority packets toward the top of the queue (step **470**). If the determination is made that the traffic within the router is

not excessive or reorganizing the queue would probably not provide greater throughput to high priority packets, then "no" branch 475 is taken bypassing the queue reorganization step. In some embodiments, the queue may be continually
5 organized to provide a weighted preference to high priority packets in order to provide greater throughput to such packets. In such a case, decision 460 could be eliminated and incoming packets automatically inserted in the queue based upon their priority. The queue could be continually
10 organized based upon packets' priority and the amount of time the packets have waited on the queue. In any event, processing continues by returning (loop 450) back to the beginning of the router processing in order to continue managing the router's queue.

15 **Figure 5** shows a block diagram of a client computer system using a pricing daemon to notify the user of the current service upgrade price. Client computing device 500 may be a stand alone PC, a handheld device, or any device capable of connecting to a computer network, such as the
20 Internet. Client computing device 500 includes browser interface 505 for communicating with the computer network. When the user wishes to perform a service upgrade, network pricing daemon 510 intercepts the request. Network pricing daemon 510 receives pricing information 515 which is
25 received from a network service provider, such as Internet Service Provider 520. Pricing data 525 is maintained by Internet Service Provider 520 so that the prices for service may be altered as needed. Network pricing daemon 510 uses the received pricing information to display
30 pricing interface 530 to the user. Based on the pricing information provided, the user decides whether to request

the service upgrade in light of the service charge that will be applied. Client decision 535 is forwarded to Internet Service Provider 520. If the client wishes to perform the service upgrade, Internet Service Provider 520 keeps track of the user's upgrade charges in billing records 540. The client uses browser 505 to send and receive both priority packets 550 and normal packets 560. Packets addressed to or from client computing device 500 are marked as priority packets 550 for faster delivery through network 570. If the user does not wish to pay for upgraded service, packets addressed to or from client computing device 500 are marked as normal priority packets 560 for normal delivery through network 570 and billing records 540 are maintained accordingly.

Figure 6a shows a window interface for a user accepting or rejecting a service upgrade. Window 600 is displayed for the user when the user requests a priority service upgrade. Window 600 instructs the user to accept or reject the service upgrade based on the current pricing charged for the service upgrade. Current pricing 605 is received from the network provider, or Internet Service Provider, and displayed on window 600 for user action. The user selects "yes" command button 610 to accept the pricing for the service upgrade. If the user selects "no" command button 615, the user's service is not upgraded and the user is not charged for priority service.

Figure 6b shows a flowchart for daemon processing for accepting or rejecting service upgrade pricing. Daemon processing begins at 620 whereupon a decision is made whether the user has requested a network service upgrade (decision 625). If the user has not requested a network

service upgrade, "no" branch 626 is repeatedly taken looping back to the beginning of the processing until the user requests a network service upgrade. Daemon processing can also be programmed to wait for the user to request a service upgrade by using a semaphore or other type of interrupt to indicate that the user wishes to upgrade his or her network service. When the user requests to upgrade the network service, "yes" branch 628 is taken whereupon the current upgrade pricing is requested from the network service provider (step 630). This pricing information is received and an appropriate dialog window is displayed for the user including the current pricing information (output 640, see **Figure 6a** for an example dialog box). The user's selection is received (input 650) based on the information provided to the user. If the pricing is accepted by the user, decision 660 branches to "yes" branch 665 whereupon the network service provider is notified (output 670) to begin marking the user's packets with priority headings and charging the user for priority service. On the other hand, if the pricing is not accepted by the user, decision 660 branches to "no" branch 675 whereupon the network service provider is not notified that the user wishes to upgrade service and normal pricing and service levels are maintained (step 680). Additionally, if the user was currently in an upgrade mode, step 680 would operate to reset the user's priority and pricing to normal levels with the network service provider. After the user's selection has been processed, daemon processing ends at 690. Note that daemon processing may be iterative to provide continuous monitoring of the user's service requests, however **Figure 6b** illustrates the processing of one such service request.

Figure 7 illustrates network pricing server 700 determining pricing levels for upgraded service as well as handling client requests regarding upgraded service. Network pricing server 700 includes pricing engine 720 for determining current pricing levels which are stored in pricing data store 735. Pricing engine 720 is a process, such as a daemon, that issues network traffic inquiries 725 to one or more devices included in computer network 775. An example computer network is the Internet. Network devices that receive network traffic inquiries include routers, switches, and servers that handle the flow of requests and responses through computer network 775. Traffic data 730 is returned from these devices. A pricing table or pricing algorithm is used by pricing engine 720 to determine the current service upgrade price or prices that are stored in pricing data store 735.

Client computer 780 issues pricing inquiry request 785 in order to view the current upgrade pricing. Pricing inquiry request 785 travels through computer network 775 to network pricing server 700 as price request 705. Network pricing server 700 receives price request 705 which identifies the client's computer address (i.e., IP address). Price inquiry engine 710 is used to retrieve pricing data 735 corresponding to the client's request. The retrieved pricing data is packaged in price response 715 which is returned to client computer 780 via computer network 775. The pricing data is received by client computer 780 as pricing information 790 and displayed on the client's display to the user. Pricing inquiry 785 and pricing information 790 can be sent and received using a standard browser software application, such as Microsoft

Internet Explorer™ or Netscape Navigator™. The client's requests may be directed to a particular port address on network pricing server 700 so that the appropriate pricing software can handle the client's request.

5 One of the things that the client may decide to do in response to receiving pricing information 790 is request a service level change to increase the priority of packets sent and received by client computer 780. In order receive higher priority service, client computer 780 sends service level change request 795 through computer network 775 to network pricing server 700. Network pricing server 700 receives service change request 740 from computer network 775. Billing engine software 745 processes service change request 740. In one embodiment, price information returned in price information 790 is packaged with service change request 740 and used for billing information. In another embodiment (or if the pricing information is not included), billing engine 745 retrieves pricing data from pricing data store 735. A record is written to billing records data store 750 identifying the client and recording the pricing in effect during the upgraded service period and the timestamp when the upgrade service request was received and processed. Packets with the client as either the destination (recipient) or target (sender) are marked with priority heading information so that they travel in a prioritized fashion through network 775. When the client no longer wishes to use and pay for prioritized service, another service level change request 795 is sent by client through network 775 and received by network pricing server 700 as service change request 740, this time informing network pricing server 700 to stop the prioritized service.

Billing engine 745 responds by writing data to billing records data store 750 recording the timestamp at which the client stopped using the prioritized service and by resetting the priority heading information so that the client's packets travel in a non-prioritized (normal) fashion through network 775.. At the end of a billing cycle, billing records 750 are processed to calculate client bills including the amount owed for prioritized service during the billing period. This calculation is made by multiplying the pricing applicable to the client's service upgrade requests by the amount of time the client used the corresponding service upgrades.

Figure 8 illustrates a flowchart for determining the pricing levels to charge clients for service upgrade requests. Processing commences at 800 whereupon one or more traffic requests are sent to external network devices, such as routers, that handle flow through the computer network (step 810). Traffic requests 815 are received by external network devices 820 which respond by returning traffic data 825 indicating the amount of traffic passing through the various devices. Traffic data is received from the external network devices (step 830) and used to calculate the current amount to charge for upgraded service based on the traffic levels (step 840). Alternatively, or in addition to, the traffic data other information such as time of day and day of the week information can be used to determine pricing data. For example, pricing during the weekend can be reduced from that during the week regardless of the amount of current network traffic. The exact formulation to use for pricing depends on the ISP or server and the goal or intention of the provider.

The calculated pricing data is stored (step 850) in pricing data store 860 so that the information is available to provide to other processes (see **Figures 7 and 9** for examples). A wait state is used (step 870) so that the pricing information is performed on a periodic interval (i.e., every 5 minutes, every hour, etc.). A determination is made as to whether processing should continue (decision 880). If processing continues, decision 880 branches to "yes" branch 885 which loops back to get the next set of traffic data. This looping continues until processing ends at which time decision 880 branches to "no" branch 890 and processing ends at 895.

Figure 9 illustrates a flowchart for handling client requests regarding upgraded network service. Processing commences at 900 whereupon request 910 is received (step 905) from client 915. A determination is made as to whether the client is inquiring about current upgrade pricing (decision 920). If the client is inquiring about current upgrade pricing, decision 920 branches to "yes" branch 922 whereupon pricing data is retrieved (step 925) from pricing data store 930 stored in a storage area. The retrieved pricing information is packaged and sent back to client 940 using the address information received in request 910 (step 935).

If the client is not inquiring about current upgrade pricing, decision 920 branches to "no" branch 942 whereupon the client's service request is parsed and processed (step 945). A determination is made as to whether the client is requesting to start using upgraded network service (decision 950). If the user is not requesting to start using upgraded network service (i.e., the request is to

stop using upgraded service), decision 950 branches to "no" branch 952 and the stop time is written to client's billing records 980 and the server (ISP) stops marking the client's packets for prioritized handling (step 955). On the other hand, if the request is to start prioritized service, decision 950 branches to "yes" branch 958 whereupon a determination (decision 960) is made as to whether a price quote was included in the client's upgrade request (the price quote being the results of a prior price check by the client). If price data is not included (or another exception occurs such as the price quote having an old timestamp or being invalidated), decision 960 branches to "no" branch 962 whereupon current pricing information is read (step 965) from pricing data store 930. On the other hand, if valid price quote data is included with the service upgrade request, decision 960 branches to "yes" branch 968 and the included price information is extracted from the client's request (step 970). A timestamp corresponding to the upgrade service request is written to client's billing records 980 along with the price that was quoted or read and the server or ISP begins prioritized handling of the client's packets until the client requests to stop the prioritized marking with a subsequent stop request.

A determination is made as to whether additional client requests need to be processed (decision 985). If there are (or will be) additional client requests to process, decision 985 branches to "yes" branch 990 which loops back to handle the next request. This looping continues until processing ends and no more requests are

handled at which time decision **985** branches to "no" branch **992** and processing ends at **995**.

In some embodiments the dynamic price in effect when the client requested a service upgrade is applied to the client's entire network session. However, in other embodiments, the network price changes periodically based on increased or decreased network demands. In these embodiments, users may request to be informed if the price changes by a certain amount. For example, a user could request to be informed if the price increases more than 20% during the user's session. If the price per minute is calculated at \$.05 per minute at the beginning of the session then the user would be informed if the price reaches \$.06 per minute. Upon being notified, the user would have the option to accept the increased price and continue the high priority network session or decline the increased price and return to normal (non prioritized) service.

Figure 10 illustrates information handling system **1001** which is a simplified example of a computer system capable of performing the server and client operations described herein. Computer system **1001** includes processor **1000** which is coupled to host bus **1005**. A level two (L2) cache memory **1010** is also coupled to the host bus **1005**. Host-to-PCI bridge **1015** is coupled to main memory **1020**, includes cache memory and main memory control functions, and provides bus control to handle transfers among PCI bus **1025**, processor **1000**, L2 cache **1010**, main memory **1020**, and host bus **1005**. PCI bus **1025** provides an interface for a variety of devices including, for example, LAN card **1030**. PCI-to-ISA bridge **1035** provides bus control to handle transfers between PCI

bus 1025 and ISA bus 1040, universal serial bus (USB) functionality 1045, IDE device functionality 1050, power management functionality 1055, and can include other functional elements not shown, such as a real-time clock (RTC), DMA control, interrupt support, and system management bus support. Peripheral devices and input/output (I/O) devices can be attached to various interfaces 1060 (e.g., parallel interface 1062, serial interface 1064, infrared (IR) interface 1066, keyboard interface 1068, mouse interface 1070, and fixed disk (HDD) 1072) coupled to ISA bus 1040. Alternatively, many I/O devices can be accommodated by a super I/O controller (not shown) attached to ISA bus 1040.

BIOS 1080 is coupled to ISA bus 1040, and incorporates the necessary processor executable code for a variety of low-level system functions and system boot functions. BIOS 1080 can be stored in any computer readable medium, including magnetic storage media, optical storage media, flash memory, random access memory, read only memory, and communications media conveying signals encoding the instructions (e.g., signals from a network). In order to attach computer system 1001 to another computer system to copy files over a network, LAN card 1030 is coupled to PCI bus 1025 and to PCI-to-ISA bridge 1035. Similarly, to connect computer system 1001 to an ISP to connect to the Internet using a telephone line connection, modem 1075 is connected to serial port 1064 and PCI-to-ISA Bridge 1035.

While the computer system described in Figure 10 is capable of executing the invention described herein, this computer system is simply one example of a computer system.

Those skilled in the art will appreciate that many other computer system designs are capable of performing the invention described herein.

One of the preferred implementations of the invention
5 is an application, namely, a set of instructions (program code) in a code module which may, for example, be resident in the random access memory of the computer. Until required by the computer, the set of instructions may be stored in another computer memory, for example, on a hard
10 disk drive, or in removable storage such as an optical disk (for eventual use in a CD ROM) or floppy disk (for eventual use in a floppy disk drive), or downloaded via the Internet or other computer network. Thus, the present invention may be implemented as a computer program product for use in a
15 computer. In addition, although the various methods described are conveniently implemented in a general purpose computer selectively activated or reconfigured by software, one of ordinary skill in the art would also recognize that such methods may be carried out in hardware, in firmware,
20 or in more specialized apparatus constructed to perform the required method steps.

While particular embodiments of the present invention have been shown and described, it will be obvious to those skilled in the art that, based upon the teachings herein,
25 changes and modifications may be made without departing from this invention and its broader aspects and, therefore, the appended claims are to encompass within their scope all such changes and modifications as are within the true spirit and scope of this invention. Furthermore, it is to
30 be understood that the invention is solely defined by the

appended claims. It will be understood by those with skill in the art that if a specific number of an introduced claim element is intended, such intent will be explicitly recited in the claim, and in the absence of such recitation no such
5 limitation is present. For a non-limiting example, as an aid to understanding, the following appended claims contain usage of the introductory phrases "at least one" and "one or more" to introduce claim elements. However, the use of such phrases should not be construed to imply that the
10 introduction of a claim element by the indefinite articles "a" or "an" limits any particular claim containing such introduced claim element to inventions containing only one such element, even when the same claim includes the introductory phrases "one or more" or "at least one" and
15 indefinite articles such as "a" or "an"; the same holds true for the use in the claims of definite articles.

T09299" 222650